

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 01-217635
 (43)Date of publication of application : 31.08.1989

(51)Int.Cl. G06F 9/46

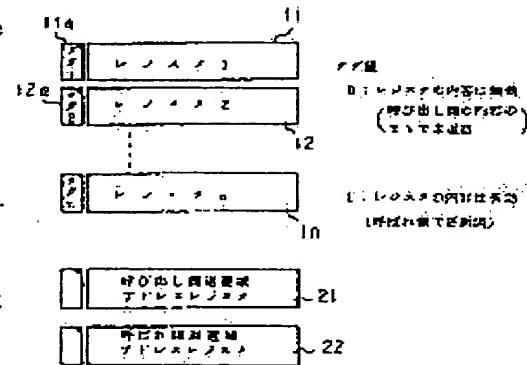
(21)Application number : 63-043793 (71)Applicant : FUJITSU LTD
 (22)Date of filing : 26.02.1988 (72)Inventor : KAWABATA YUTAKA

(54) REGISTER SAVING SYSTEM

(57)Abstract:

PURPOSE: To ensure the quick switch of tasks by saving and restoring a register with the use of a tag bit and a saving area address register and using the bus idle time during the execution of a switched task or a procedure or every time register reference and replacement instructions are issued.

CONSTITUTION: The register reference/replacement instructions are carried out after a called task is started. In this case, the present register contents are saved into a saving area pointed by a calling side saving area address register 21 as long as the tag bit of the relevant register is equal to 0. In addition, the data are read out of a saving area pointed by a called side saving area address register 22 and the register is restored. In such a way, the register is not saved nor restored at the switch of registers but the tasks are first switched. Thus the register is saved and restored by means of an idle bus during the run of the switched task or at the time point when the register is referred to and replaced in a task execution mode. As a result, the tasks are switched quickly.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

⑩ 日本国特許庁 (JP)

⑪ 特許出願公開

⑫ 公開特許公報 (A)

平1-217635

⑬ Int. Cl.
G 06 F 9/46

識別記号
313

庁内整理番号
A-7056-5B

⑭ 公開 平成1年(1989)8月31日

審査請求 未請求 請求項の数 2 (全5頁)

⑮ 発明の名称 レジスタ退避方式

⑯ 特願 昭63-43793

⑯ 出願 昭63(1988)2月26日

⑰ 発明者 川端 豊 神奈川県川崎市中原区上小田中1015番地 富士通株式会社
内

⑰ 出願人 富士通株式会社 神奈川県川崎市中原区上小田中1015番地

⑰ 代理人 弁理士 青柳 稔

明細書

1. 発明の名称

レジスタ退避方式

2. 特許請求の範囲

1. タスク切替え、割込み、または手続き呼び出し時の中央処理装置のレジスタのデータ退避、復元方式において

各レジスタ (11, 12, ...) にタグビット (11a, 12a, ...) を設け、また呼び出し側退避域アドレスレジスタ (21) と呼ばれ側退避域アドレスレジスタ (22) を設け、

これらのタグビット及び退避域アドレスレジスタを用いてレジスタ退避、復元を、切換わったタスクまたは手続きの実行中にバス空き時間を利用して、またレジスタ参照、更新命令が出たときにその都度、行なうことを特徴とするレジスタ退避方式。

2. タスク切替え時に全レジスタのタグビットを0にクリアし、また呼ばれ側退避域アドレスレジスタ (22) の内容を呼び出し側退避域アドレス

レジスタ (21) へ移し、呼ばれ側退避域アドレスレジスタへは呼ばれ側のタスクの退避域アドレスをセットし、

呼ばれ側のタスクを実行中にレジスタ参照、更新命令が出たとき、当該レジスタのタグビットを見てそれが0ならば、当該レジスタの内容を呼び出し側退避域アドレスレジスタが示すメモリアドレスへ退避し、該レジスタへ呼ばれ側退避域レジスタが示すメモリアドレスのデータをセットし、該レジスタのタグビットを1にセットすることを特徴とする請求項1記載のレジスタ退避方式。

3. 発明の詳細な説明

(発明の概要)

コンピュータでタスク切替えが行なわれ、または割込みが入り、あるいは手続き (サブルーチン) を呼び出したときのレジスタの退避方式に関し、

切替えを迅速に行なうことができ、プログラム作成誤まりによりレジスタ内容を喪失したりすることがないレジスタ退避方式を提供することを目的とし、

特開平1-217635 (2)

タスク切替え、割込み、または手続き呼び出し時の中間処理装置のレジスタのデータ退避、復元方式において各レジスタにタグビットを設け、また呼び出し側退避域アドレスレジスタと呼ばれる側退避域アドレスレジスタを設け、これらのタグビット及び退避域アドレスレジスタを用いてレジスタ退避、復元を、切換わったタスクまたは手続きの実行中にバス空き時間を利用して、またレジスタ参照、更新命令が出たときにその都度、行なう構成とする。

(産業上の利用分野)

本発明は、コンピュータでタスク切替えが行なわれ、または割込みが入り、あるいは手続き（サブルーチン）を呼び出したときのレジスタの退避方式に関する。

タスク切替え、割込み、または手続き呼び出し時には、前に動作していたタスクまたは呼び出し元手続きの環境を保存するため、従来方式ではOSまたは呼ばれた手続きにおいて、プログラムの責任

で、機械語命令によって、全レジスタの退避／復元をしなければならない。

〔従来の技術〕

第5図～第7図にレジスタの退避、復元の要領を示す。第5図はタスク切替時そので、タスクA実行中にタスクBへの切替え事象が発生するとOS(Operating System)は、CPUの全レジスタの内容を読み出して退避域Aへ格納し、次いで退避域Bより、退避させてあったタスクB実行時のレジスタの内容を読み出してこれを該当レジスタへセットし、こうしてタスクB実行環境を整えた後、タスクBに実行を指示する。

第6図は割込み時のそれで、タスクA実行中に割込みが入るとOSは全レジスタの内容を退避域Aへ退避させ、割込み処理終了後、全レジスタを復元し、タスクAに再起動をかける。

第7図は手続き呼び出し時のそれで、実行中の手続きAが手続きBを呼び出すと、手続きBは全レジスタの内容を退避域Aへ退避させ、然る後手続き

Bの実行に入り、その実行完了で、退避域Aを読み出して全レジスタの復元を行ない、手続きAの実行継続へ復帰させる。

しかしこの従来方式には次のような問題がある。即ち、①レジスタの退避、復元に時間がかかる。これはレジスタの数の多いCPUほど著しい。レジスタの退避、復元に要する時間は、特に割込み処理においては応答時間に影響するため極めて重大である。また②レジスタの退避、復元がプログラムの責任でなされるため、プログラムに誤りがあると（プログラムが退避、復元の命令群を書き加えるのを忘れたりすると）レジスタの内容が保証されず、自タスクまたは自手続きばかりでなく他タスクまたは他手続きの動作まで妨害することにもなる。

(発明が解決しようとする課題)

本発明はかかる点を改善し、切替えを迅速に行なうことができ、プログラム作成誤りによりレジスタ内容を喪失したりすることがないレジスタ

退避方式を提供することを目的とするものである。

〔課題を解決するための手段〕

第1図に示すように本発明ではCPU(中央処理装置)のレジスタ11, 12, ……にタグビット11a, 12a, ……を付加し、また呼び出し側退避域アドレスレジスタ21および呼ばれ側退避域アドレスレジスタ22を設け、これらによりレジスタの退避、復元をCPUが自動的に行なうようにする。

〔作用〕

CPUはタグビット11a, 12a, ……および退避域アドレスレジスタ21, 22を用いて、次のような動作でレジスタ退避を行なう。

①タスク切替時に全レジスタのタグビットを0にクリアする。また退避域アドレスレジスタを更新する。即ち、呼ばれ側退避域アドレスレジスタ22の内容を呼び出し側退避域アドレスレジスタ21に移し、前者（レジスタ22）には今呼び

特開平1-217635 (3)

出そうとしているタスクの退避域アドレスを格納する。

②呼び出されたタスクが起動した後、レジスタを参照／更新する命令を実行するとき、そのレジスタのタグビットを見てそれが0（レジスタの内容は無効、未退避で内容は呼び出し側のときのまゝ）ならば現在のレジスタ内容をレジスタ21の指す退避域に退避し、更にレジスタ22の指す退避域からデータを読出してきてレジスタを復元する（この復元処理は、手続き呼び出しの場合は不要）。復元したら、レジスタのタグビットを1（レジスタの内容は有効、即ち更新済みでありレジスタ内容は呼び出し側のそれ）にセットする。

③上記②以前にも、呼び出されたタスクが起動したその命令の実行と並行して、バスの空き時間を利用して自動的に②の操作即ちタグが0のレジスタについての退避、復元を行なう。

このように本発明ではタスク切替時にレジスタ退避、復元を行なうのではなく、先ずタスク切替を行ない、切替ったタスクが走行中にバスの空

きを利用してまた該タスク実行でレジスタ参照、更新があればその時点でレジスタ退避、復元を行なうので、タスク切替を迅速に行なうことができる。バス空き時間を利用してレジスタ退避、復元が済んでしまえば、これに要する時間は表面に現われなくなり、無いのと同じになる。

また退避、復元はCPUが自動的に行なうので、プログラム作成誤まりによってレジスタの退避、復元ができず動作に支障を与えることはない。

(実施例)

第2図にタスク切替（または割込み、または手続き呼び出し、を含むが、これらは適宜省略する）命令実行時のレジスタ退避、復元に関する処理を示す。この処理では①全レジスタのタグを0にクリアし、②呼び出し側退避域レジスタXの内容を呼び出し側退避域レジスタYの内容に切替える。タスクAをタスクBに切替える場合を例にすると、タスクAの退避域アドレスはレジスタYに入っている、上記②の処理でそれがレジスタXに移され

る。次に③レジスタYへタスクBの退避域アドレスをセットし、これでタスク切替時の処理は終りである。

第3図は切替ったタスクが実行中にレジスタ参照、更新命令が出たときの、レジスタ退避、復元に関する処理を示す。①実行対象の命令があるレジスタに対する使用（参照／更新）命令か否かチェックし、使用命令であれば②そのタグは0か否かチェックし、0なら③レジスタXが示すアドレスへ当該レジスタの内容を退避させ、代って④該レジスタへレジスタYが示すアドレスのメモリ内容をセットし、⑤該レジスタのタグを1にセットする。これでレジスタの退避、復元が完了し、当該命令の実行に移る。

この第3図の処理はレジスタ参照／更新命令が出る毎に、従って各レジスタ毎に行なわれる。図示しないが、バスに空き時間があればその時も（レジスタ退避／更新命令が出ていなくても）各レジスタにつき逐次、退避／復元が行なわれ、タグビットを0から1に切替える。レジスタ退避／更

新命令でまたはバス空きを利用してかかる処理が行なわれ、それが全レジスタに及んだとき、レジスタ退避／復元処理は終了である。退避、復元用アドレスはレジスタX、Y（第1図の21、22）にある。これはレジスタ群に対する共通のものであるが、その中の個々のレジスタに対するアドレスは、第1レジスタ、第2レジスタ、……などの順番を用いることで固定的に決定される。

第4図はレジスタ退避／復元例を示す。レジスタは1つ（11）のみ示すが、他のレジスタ12、……1nについても同様である。①タスク切替前にレジスタ11の内容が1111、タグビットは1、レジスタ21の内容は2222、レジスタ22の内容は2222であったとし、タスクAが実行中でこれがタスクBに切替わるとする。レジスタ22の内容3333はタスクAの退避域アドレスで、メモリ30のこのアドレスに本例では9999なるデータが入っている。

②タスク切替命令実行で、レジスタ11のタグビットは0にクリアし、レジスタ21にレジスタ

特開平1-217635 (4)

22の内容3333をセット、そしてレジスタ22にタスクBの退避域アドレス5555をセットする。

(b)レジスタ使用命令実行で、レジスタ11の内容1111はレジスタ21が示すメモアドレス3333へストアされ、レジスタ11へはタスクBの退避域アドレス5555のメモリデータ6666がセットされ、レジスタ11のタグビットは1にセットされる。

図示しないが、タスクBから更にタスクCに切替わるとときは第4回向を両間にした状態となり、次のステップでレジスタ11のタグビットを0にクリア、レジスタ21にレジスタ22の内容5555をセット、レジスタ22にタスクCの退避域のアドレスをセットし、レジスタ使用命令実行でレジスタ11の内容6666をレジスタ21が示すアドレス5555へ退避、レジスタ11へはレジスタ22が示すアドレスのデータをセットする。

手続き呼び出しではレジスタ退避はするが、復元(呼び出された手続きのレジスタデータのレジス

タへのセット)はしないから、第2回のステップ3および第3回のステップ③は不要である。

〔発明の効果〕

以上説明したように本発明では、タスク切替時にはレジスタの退避、復元は行なわないので、切替え時間を短縮することができる。バスの空き時間を利用してレジスタの退避、復元をするため、最高では従来の退避、復元命令の実行時間だけタスク切替が速くなる。バスが空かず、レジスタ参照/更新命令でのレジスタ退避/復元を行なった場合が最悪であるが、この場合でも所要時間は従来と同じである。

また本発明では退避、復元はCPUが自動的に行なうため、プログラムの誤りによって他のタスク(他手続き)に影響を与えることがない。

4.図面の簡単な説明

第1図は本発明の原理説明図、

第2図はタスク切替時の処理を示す流れ図、

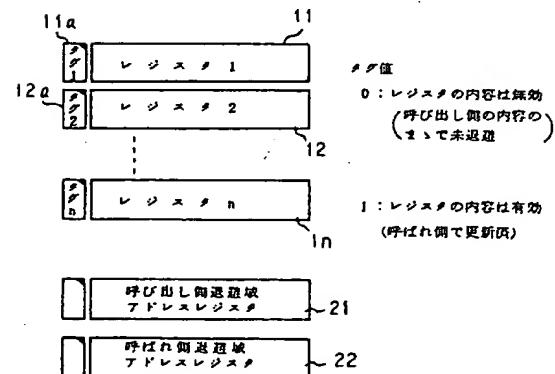
第3図はレジスタ退避、復元動作を示す流れ図、

第4図はレジスタ退避/復元処理例の説明図、

第5図～第7図は従来のレジスタ退避、復元の説明図で第5図はタスク切替時、第6図は割込み時、そして第7図は手続き呼び出し時である。

第1図で11, 12, ……はCPUのレジスタ、11a, 12a, ……はそのタグビット、21は呼び出し側退避域アドレスレジスタ、22は呼ばれ側退避域アドレスレジスタである。

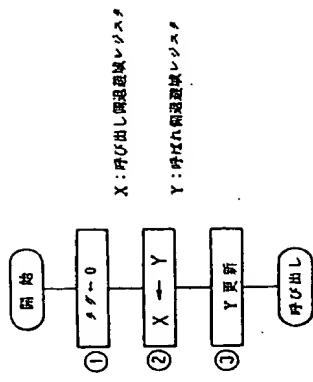
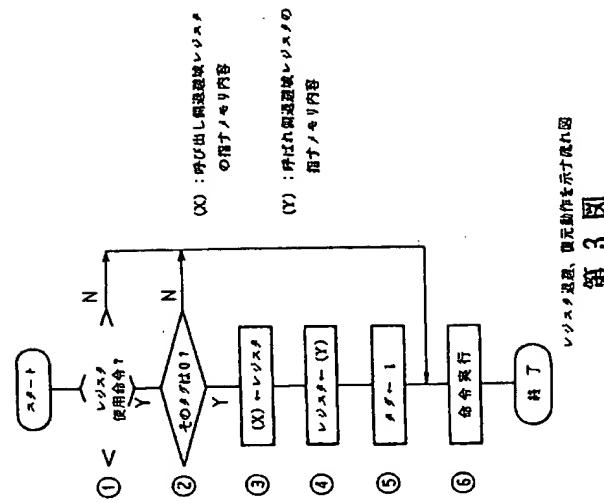
出願人 富士通株式会社
代理人弁理士 青柳 桂



本発明の原理説明図

第1図

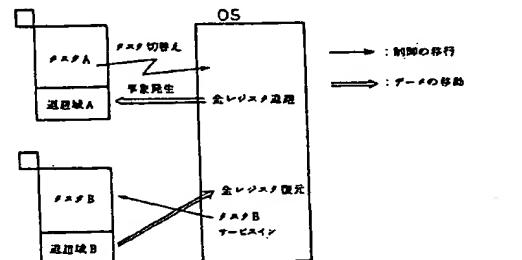
特開平1-217635 (5)

メモリ切替時の処理を示す流れ図
第2図レジスト処理、現元動作を示す流れ図
第3図

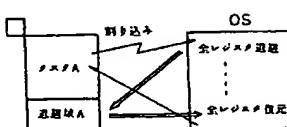
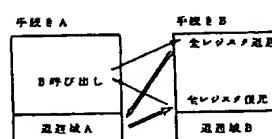
(a) レジスト切替		30:49	
11a	レジスト	2222	0000
11	呼び出し側	3333	9999
21	呼び出し側	2222	0000
22	呼び出し側	5555	6666
		3333	9999
		2222	0000
		5555	6666

(b) レジスト切替命令		30:49	
CHGTSK	5555	2222	0000
実行後		3333	9999
21	3333	2222	0000
22	5555	3333	9999
		5555	6666

(c) レジスト使用命令		30:49	
実行後		2222	0000
11	6666	3333	1111
21	3333	2222	0000
22	5555	5555	6666

レジスト追跡/現元動作例の説明図
第4図メモリ切替時のレジストの
追跡・現元の説明図
第5図

第5図

割込み時のレジストの
追跡・現元の説明図
第6図手続を呼び出し時のレジストの
追跡・現元の説明図
第7図